UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/679,434	10/03/2000	Peter C. Damron	SUN-P5369	7010

7590 03/23/2004

Russ F Marsden
Sierra Patent Group Ltd
PO Box 6149
Stateline, NV 89449

EXAMINER	
SHRADER, LAWRENCE J	
ART UNIT	**PAPER NUMBER**
2124	3

DATE MAILED: 03/23/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/679,434	Applicant(s) DAMRON ET AL.
	Examiner Lawrence Shrader	Art Unit 2124

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *03 October 2000*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-18* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-18* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *#2, 1/22/2001*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Receipt of the Information Disclosure Statement on 1/22/2001 is acknowledged.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter, which the applicant regards as his invention.

The term "close to" in claim 3c is a relative term, which renders the claim indefinite. The term "close to" is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention.

Specification

3. The abstract of the disclosure is objected to because the length exceeds 150 words. Correction is required. See MPEP § 608.01(b).

Applicant is reminded of the proper language and format for an abstract of the disclosure.

The abstract should be in narrative form and generally limited to a single paragraph on a separate sheet within the range of 50 to 150 words. It is important that the abstract not exceed 150 words in length since the space provided for the abstract on the computer tape used by the printer is limited. The form and legal phraseology often used in patent claims, such as "means" and "said," should be avoided. The abstract should describe the disclosure sufficiently to assist readers in deciding whether there is a need for consulting the full patent text for details.

The language should be clear and concise and should not repeat information given in the title. It should avoid using phrases which can be implied, such as, "The disclosure concerns," "The disclosure defined by this invention," "The disclosure describes," etc.

4. The Applicant is requested to update the serial number information of the related applications disclosed on page 1 of the specification.

5. Claims 4, 11, and 12 are objected to because of the following informalities:

Claims 4 and 11 both have two step (c)'s that should be corrected to read 4(d) and 11(d) respectively; also claim 12 has two step (a)'s, therefore claim 12 should be re-lettered with steps a through g. Appropriate correction is required.

Claim Rejections - 35 USC § 102

6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

7. Claims 1, 7; 8, 14, and 15 are rejected under 35 U.S.C. 102(b) as being anticipated by Rasbold et al., U.S. Patent 5,202,975 (hereinafter referred to as Rasbold).

In regard to claim 1:

Rasbold discloses a scheduler with a method for ordering instructions:

"(a) determining dependencies between instructions in said plurality of instructions;"

Rasbold discloses the determining of dependencies at column 2, lin55 – 58.

"(b) creating a directed acyclic graph showing said dependencies in said plurality of instructions, where said directed acyclic graph's nodes each correspond to an instruction from said plurality of instructions;"

Rasbold discloses the creation of a directed acyclic graph (DAG) where the nodes correspond to one of a plurality of instructions (column 8, lines 59 – 62; column 3, lines 16 – 20; e.g., Figure 2)

"(c) creating at least one queue;"

Rasbold discloses the creation of at least one queue (column 2, lines 1 – 6).

"(d) traversing said directed acyclic graph in a dependency-preserving manner;"

Rasbold discloses traversing the directed acyclic graph in a dependency-preserving manner (column 2, lines 55 – 58; column 3, lines 15 – 29).

"(e) creating a ready set of nodes by identifying which nodes in said directed acyclic graph are in a ready state and which instructions correspond to said nodes;"

Rasbold discloses a ready set of instructions (corresponding to the DAG nodes) that are scheduled for issuance (column 4, lines 20 – 27; column 9, lines 50 – 65).

"(f) finishing if there are no nodes found to be in a ready state;"

Rasbold discloses that the scheduling is finished when the ready set has no more instructions (nodes) to process (column 10, lines 24 – 29).

"(g) identifying said at least one queue's needs to have queue elements added in accordance with said at least one target processor;"

Rasbold discloses the addition of instructions into a queue that wait to be issued to a processor (column 2, lines 1 – 6). Also see Figure 3 refs. 46 – 50.

"(h) identifying said at least one queue's needs to have queue elements subtracted in accordance with said at least one target processor;"

Rasbold discloses the identification of an instruction that must have its issuance delayed (subtracted from an ordering) by pushing (enqueueing) the issuance of the instruction back in order (that is back in some form of queue) allowing another instruction that will benefit from early issuance to be moved to the ready set (column 4, lines 51 – 60).

"(i) choosing one of said nodes and its corresponding instruction in said ready set and one of said at least one queues such that said chosen node and said queue satisfy at least one of said queue's needs;"

Rasbold discloses a queue that issues instructions (corresponding to a DAG node) at column 2, lines 1 – 6. Also, disclosed at column 4, lines 51 – 60 is that instructions in the ready set are scheduled for execution, therefore the ready set satisfies the needs of the scheduling queue.

"(j) choosing one of said nodes and its corresponding instruction in said ready set and one of said at least one queues heuristically if no node can satisfy at least one of said queue's needs;"

Rasbold discloses that if an instruction (corresponding to a node) is heuristically chosen if it can benefit form an early execution when another instruction issuance is delayed; it is moved to the ready set to "bubble" to the top of the queue of instructions to be scheduled (column 4, lines 51 – 60).

"(k) removing said chosen node from said directed acyclic graph;"

See Figure 4, ref. 48 of Rasbold and the accompanying text at column 10, lines 14 - 23.

"(1) modifying said chosen queue in accordance with said chosen node and its associated instruction;"

Rasbold discloses a queue issuing instructions (corresponding to a DAG node) at column 2, lines 1 – 6. The ready instructions are scheduled in the queue (see Figure 4, ref. 46) as new instructions enter or are shifted (modifying the queue) as disclosed at step (j) above.

"(m) modifying the order of instructions in said code file in accordance with said chosen node and its associated instruction; and,"

Rasbold discloses that if an instruction (corresponding to a node) is heuristically chosen if it can benefit form an early execution when another instruction issuance is delayed; it is moved to the ready set to "bubble" to the top of the queue of instructions to be scheduled (column 4, lines 51 – 60).

"(n) continuing processing at (d)."

See Figure 4 of Rasbold.

In regard to claim 7, incorporating the rejection of claim 1:

"...replacing modifying said chosen queue, the method further comprising:

(a) adding an element corresponding to said chosen node to said at least one queue if said at least one queue's needs is to have queue elements added; and,"

Rasbold discloses the addition of instructions into a queue that wait to be issued to a processor (column 2, lines 1 – 6). Also see Figure 3 refs. 46 – 50.

"(b) removing at least one element in accordance with said chosen node to said at least one queue if said at least one queue's needs is to have queue elements removed."

Rasbold discloses the identification of an instruction that must have its issuance delayed (subtracted from an ordering) by pushing the issuance of the instruction back in order (that is back in some form of queue) allowing another instruction that will benefit from early issuance to be moved to the ready set (column 4, lines 51 – 60).

In regard to claim 8 (a storage device), it is rejected for the same corresponding reasons put forth in the rejection of claim 1 (a corresponding method).

In regard to claim 14, incorporating the rejection of claim 8 (a storage device), it is rejected for the same corresponding reasons put forth in the rejection of claim 7 (a corresponding method).

In regard to claim 15:

> *"A queue modeling instruction scheduler apparatus for use in compiling a program, the apparatus executable in a device having a processor operatively coupled to a memory, the apparatus comprising:*
>
> *a directed acyclic graph creation module operatively disposed within said apparatus, configured to determine dependencies between instructions in a program to be compiled and to create a directed acyclic graph showing said dependencies in said program, where said directed acyclic graph's nodes correspond to instructions in said program; "*

An apparatus corresponding to the method of claim 1. Rasbold discloses the creation of a directed acyclic graph (DAG) where the nodes correspond to one of a plurality of instructions (column 8, lines 59 – 62; column 3, lines 16 – 20; e.g., Figure 2).

> *"a directed acyclic graph traversal and ready set identification module operatively disposed within said apparatus and configured to traverse said directed acyclic graph in a dependency-preserving manner and to create a ready set of nodes;"*

An apparatus corresponding to the method of claim 1. Rasbold discloses traversing the directed acyclic graph in a dependency-preserving manner (column 2, lines 55 – 58; column 3, lines 15 – 29). Rasbold also discloses a ready set of instructions (corresponding to the DAG nodes) that are scheduled for issuance (column 4, lines 20 – 27; column 9, lines 50 – 65).

"a ready set evaluation module operatively disposed within said apparatus and configured to identify which nodes in said ready set correspond to which instructions in said program, and to evaluate said instructions for their effect on memory operations;"

An apparatus corresponding to the method of claim 1. Rasbold also discloses a ready set of instructions (corresponding to the DAG nodes) that are scheduled for issuance (column 4, lines 20 – 27; column 9, lines 50 – 65).

"a queue management module operatively disposed within said apparatus and configured to manage at least one queue, where managing a queue further comprises adding and removing elements from said at least one queue where said elements correspond to nodes from said directed acyclic graph;"

An apparatus corresponding to the method of claim 1. Rasbold discloses the addition of instructions into a queue that wait to be issued to a processor (column 2, lines 1 – 6). Also see Figure 3 refs. 46 – 50. Rasbold discloses the identification of an instruction that must have its issuance delayed (subtracted from an ordering) by pushing (enqueueing) the issuance of the instruction back in order (that is back in some form of queue) allowing another instruction that will benefit from early issuance to be moved to the ready set (column 4, lines 51 – 60).

"a code scheduling module operatively disposed within said apparatus and operably connected to said program and said directed acyclic graph traversal and ready set identification module and said ready set evaluation module and said queue management module and said directed acyclic graph, configured to add and remove nodes from said directed acyclic graph and to have elements of said at least one queue added and removed and to change the order of instructions in said program in accordance with said instructions, said nodes, and said at least one queue."

An apparatus corresponding to the method of claim 1. Rasbold discloses a queue issuing instructions (corresponding to a DAG node) at column 2, lines 1 – 6. The ready instructions are scheduled in the queue (see Figure 4, ref. 46) as new instructions enter or are shifted (modifying the queue) as disclosed at step (1j) above. Rasbold discloses that if an instruction (corresponding

to a node) is heuristically chosen if it can benefit form an early execution when another instruction issuance is delayed; it is moved to the ready set to "bubble" to the top of the queue of instructions to be scheduled (column 4, lines 51 – 60).

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. Claims 2, 3; 9, and 10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rasbold et al., U.S. Patent 5,202,975 as applied to claim 1 above, and further in view of Cohen, U.S. Patent 5,881,315.

In regard to claim 2, incorporating the rejection of claim 1:

"...after creating at least one queue, the method further comprising:

(a) using one queue for said at least one queue;"

Rasbold discloses the creation of at least one queue (column 2, lines 1 – 6).

"(b) determining a maximum desirable number of elements identifier for said one queue;"

Rasbold discloses the use of a queue, but does not disclose an identifier for the maximum number of elements. However, Cohen discloses an identifier ("qmax") for the maximum desired queue size. Therefore, it would have been obvious to one skilled in the art at the time the

invention was made to modify the instruction queue as taught by Rasbold with the maximum desired queue size identifier of Cohen, because the added feature allows the Rasbold invention to monitor the size of the scheduler queue in order to prevent compound effects and latency problems as taught by Rasbold at column 2, lines 55 – 63.

> *"(c) choosing one of said nodes in said ready set that will add an element to said queue if said queue has fewer elements than said identifier;"*

Rasbold discloses the addition of instructions into a queue that wait to be issued to a processor (column 2, lines 1 – 6; also see Figure 3 refs. 46 – 50), but does not disclose an identifier for the maximum number of elements. However, Cohen discloses an identifier ("qmax") for the maximum desired queue size. Therefore, it would have been obvious to one skilled in the art at the time the invention was made to add to the instruction queue as taught by Rasbold with the maximum desired queue size identifier of Cohen, because the added feature allows the Rasbold invention to monitor the size of the scheduler queue in order to prevent compound effects and latency problems as taught by Rasbold at column 2, lines 55 – 63.

> *"(d) choosing one of said nodes in said ready set that will remove at least one element from said queue if said queue has an equal number of elements as said identifier; and,"*

Rasbold discloses the identification of an instruction that must have its issuance delayed (subtracted from an ordering) by pushing the issuance of the instruction back in order (that is back in some form of queue) allowing another instruction that will benefit from early issuance to be moved from the ready set to be issued (column 4, lines 51 – 60), but does not disclose an identifier for the maximum number of elements. However, Cohen discloses an identifier ("qmax") for the maximum desired queue size. Therefore, it would have been obvious to one skilled in the art at the time the invention was made to remove an element from the instruction

queue as taught by Rasbold with the maximum desired queue size identifier of Cohen, because the added feature allows the Rasbold invention to monitor the size of the scheduler queue in order to prevent compound effects and latency problems as taught by Rasbold at column 2, lines 55 – 63.

"(e) choosing one of said nodes in said ready set that will remove at least one element from said queue if said queue has more than the number of elements as said identifier;"

Rasbold discloses the identification of an instruction that must have its issuance delayed (subtracted from an ordering) by pushing the issuance of the instruction back in order (that is back in some form of queue) allowing another instruction that will benefit from early issuance to be moved from the ready set to be issued (column 4, lines 51 – 60), but does not disclose an identifier for the maximum number of elements. However, Cohen discloses an identifier ("qmax") for the maximum desired queue size. Therefore, it would have been obvious to one skilled in the art at the time the invention was made to remove an element from the instruction queue as taught by Rasbold with the maximum desired queue size identifier of Cohen, because the added feature allows the Rasbold invention to monitor the size of the scheduler queue in order to prevent compound effects and latency problems as taught by Rasbold at column 2, lines 55 – 63.

In regard to claim 3, incorporating the rejection of claim 1:

"...after creating at least one queue, the method further comprising:

(a) using one queue for said at least one queue;"

Rasbold discloses the creation of at least one queue (column 2, lines 1 – 6).

"(b) determining a maximum desirable number of elements identifier for said one queue; "

Rasbold discloses the use of a queue, but does not disclose an identifier for the maximum number of elements. However, Cohen discloses an identifier ("qmax") for the maximum queue size. Therefore, it would have been obvious to one skilled in the art at the time the invention was made to modify the instruction queue as taught by Rasbold with the maximum queue size identifier of Cohen, because the added feature allows the Rasbold invention to monitor the size of the scheduler queue in order to prevent compound effects and latency problems as taught by Rasbold at column 2, lines 55 – 63.

> *"(c) choosing one of said nodes in said ready set such that said queue has a total number of elements close to and not exceeding said identifier;"*

Rasbold discloses the use of a queue from which an instruction (node) is issued (column 2, lines 1 – 6), but does not disclose an identifier for the maximum number of elements. However, Cohen discloses an identifier ("qmax") for the maximum queue size, which keeps the number of elements from exceeding the identifier. Therefore, it would have been obvious to one skilled in the art at the time the invention was made to modify the instruction queue as taught by Rasbold with the maximum queue size identifier of Cohen, because the added feature allows the Rasbold invention to monitor the size of the scheduler queue in order to prevent compound effects and latency problems as taught by Rasbold at column 2, lines 55 – 63.

In regard to claim 9, incorporating the rejection of claim 8 (a storage device), it is rejected for the same corresponding reasons put forth in the rejection of claim 2 (a corresponding method).

In regard to claim 10, incorporating the rejection of claim 8 (a storage device), it is rejected for the same corresponding reasons put forth in the rejection of claim 3 (a corresponding method).

10. Claims 4, 5; 11, 12; 16 and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rasbold et al., U.S. Patent 5,202,975 as applied to claim 1 above, and further in view of Eickemeyer et al., U.S. Patent 5,377,336 (hereinafter referred to as Eickmeyer).

In regard to claim 4, incorporating the rejection of claim 1:

> *"...after creating at least one queue, the method further comprising:*
>
> *(a) using a load queue, a prefetch queue, and a store queue for said at least one queue;"*

Rasbold discloses the creation of at least one queue (column 2, lines 1 – 6), which is used as a prefetch queue to keep instructions operating at the fastest and most efficient order (column 2, lines 46 – 51).

> *"(b) determining and correlating a maximum desirable number of elements identifier for each of said load queue, said prefetch queue, and said store queue;"*

Rasbold discloses the creation of at least one queue (column 2, lines 1 – 6), which is used as a prefetch queue to keep instructions operating at the fastest and most efficient order (column 2, lines 46 – 51), but does not explicitly disclose a load and store queue, nor determine and correlate a maximum desirable number of elements identifier for each queue. However, Eickemeyer discloses both a load queue, with the number of entries (elements) in the queue determined by identifying performance and cost trade-offs (maximum desirable number of elements (column 8, lines 30 - 33); and a store queue (column 11, lines 19 – 55). Therefore, it would have been obvious to one skilled in the art at the time the invention was made to modify the instruction queue as taught by Rasbold with load and store queues, as well as identifying the maximum number of element in each queue as taught by Eickemeyer, because one would be

motivated to decrease the overall execution time of Rasbold with more efficient loading and storing as taught in the Eickemeyer Abstract.

> *"(c) determining a precedence ordering between said load queue, said prefetch queue, and said store queue; and,"*

Rasbold discloses the creation of at least one queue (column 2, lines 1 – 6), which is used as a prefetch queue to keep instructions operating at the fastest and most efficient order (column 2, lines 46 – 51), but does not explicitly disclose a load and store queue. However, Eickemeyer discloses both a load queue, and a store queue (column 11, lines 19 – 55); as well as a precedence ordering illustrated at column 10. Therefore, it would have been obvious to one skilled in the art at the time the invention was made to modify the instruction queue as taught by Rasbold with load and store queues taught by Eickemeyer having precedence ordering, because one would be motivated to decrease the overall execution time of Rasbold with more efficient loading and storing as taught in the Eickemeyer Abstract.

> *"(c) choosing one of said nodes in said ready set that will change the number of elements in one of said load queue, said prefetch queue, or said store queue in accordance with said precedence order and one of said correlated identifiers."*

Rasbold discloses the identification of an instruction that must have its issuance delayed (subtracted from an ordering) by pushing (enqueueing) the issuance of the instruction back in order (that is back in the prefetch queue) allowing another instruction that will benefit from early issuance to be moved to the ready set (column 4, lines 51 – 60), but does not disclose a precedence ordering of queues, nor correlated identifiers. However, Eickemeyer discloses both a load queue, and a store queue (column 11, lines 19 – 55), as well as a precedence ordering illustrated at column 10. Therefore, it would have been obvious to one skilled in the art at the

time the invention was made to modify the instruction queue as taught by Rasbold with load and store queues taught by Eickemeyer having precedence ordering and maximun queue element identifiers, because one would be motivated to decrease the overall execution time of Rasbold with more efficient loading and storing as taught in the Eickemeyer Abstract.

In regard to claim 5, incorporating the rejection of claim 1:

"...after creating at least one queue, the method further comprising:

(a) determining which number of queues to use in accordance with a target processor;"

Rasbold discloses determining which number of queues to use in accordance with a target processor, e.g., a pipelined scalar/vector processor (column 5, lines 40 – 64).

"(a) using said determined number of queues for said at least one queue;"

Rasbold discloses using said determined number of queues to use in accordance with a target processor, e.g., a pipelined scalar/vector processor (column 5, lines 47 – 64).

"(b) determining a maximum desirable number of elements identifier for each of said determined number of queues;
(c) coupling a maximum desirable number of elements identifier to each of said determined number of queues;"

Rasbold discloses the use of a queue, but does not disclose a determining or coupling of an identifier for the maximum number of elements. However, Eickemeyer discloses both a load queue, with the number of entries (elements) in the queue determined by identifying performance and cost trade-offs (maximum desirable number of elements (column 8, lines 30 - 33); and a store queue (column 11, lines 19 – 55). Therefore, it would have been obvious to one skilled in the art at the time the invention was made to modify the instruction queue as taught by Rasbold with load and store queues, as well as identifying the maximum number of element in each

queue as taught by Eickemeyer, because one would be motivated to decrease the overall execution time of Rasbold with more efficient loading and storing as taught in the Eickemeyer Abstract.

"(d) determining a precedence ordering between each of said determined number of queues;"

Rasbold discloses the creation of at least one queue (column 2, lines 1 – 6), which is used as a prefetch queue to keep instructions operating at the fastest and most efficient order (column 2, lines 46 – 51), but does not disclose a precedence ordering between queues. However, Eickemeyer discloses a precedence ordering illustrated at column 10. Therefore, it would have been obvious to one skilled in the art at the time the invention was made to modify the instruction queue as taught by Rasbold with load and store queues taught by Eickemeyer having precedence ordering, because one would be motivated to decrease the overall execution time of Rasbold with more efficient loading and storing as taught in the Eickemeyer Abstract.

"(e) choosing one of said nodes in said ready set that will change the number of elements in one of said determined number of queues, in accordance with said precedence order and said coupled identifier, if one of said nodes can be found; and,"

Rasbold discloses the identification of an instruction (node) in the ready set that must have its issuance delayed (subtracted from an ordering – changing the number of elements) by pushing (enqueueing) the issuance of the instruction back in order (that is back in the prefetch queue) allowing another instruction that will benefit from early issuance to be moved to the ready set (column 4, lines 51 – 60), but does not disclose a precedence ordering of queues. However, Eickemeyer discloses a load queue, and a store queue (column 11, lines 19 – 55) with a precedence ordering illustrated at column 10. Therefore, it would have been obvious to one

skilled in the art at the time the invention was made to modify the instruction queue as taught by Rasbold with load and store queues taught by Eickemeyer having precedence ordering and maximun queue element identifiers, because one would be motivated to decrease the overall execution time of Rasbold with more efficient loading and storing as taught in the Eickemeyer Abstract.

> *"(f) choosing one of said nodes in said ready set that will not change the number of elements in one of said determined number of queues, in accordance with said precedence order and said coupled identifier, if none of said nodes in said ready can be found that will change the number of elements in one of said determined number of queues, in accordance with said precedence order and said coupled identifier."*

Rasbold discloses choosing an instructions when scalar interlocks are cleared, which would not change the number of elements in one of said queues (column 11, lines 66 – 67), but does not disclose a precedence order. However, Eickemeyer discloses a load queue, and a store queue (column 11, lines 19 – 55) with a precedence ordering illustrated at column 10. Therefore, it would have been obvious to one skilled in the art at the time the invention was made to modify the instruction queue as taught by Rasbold with load and store queues taught by Eickemeyer having precedence ordering, because one would be motivated to decrease the overall execution time of Rasbold with more efficient loading and storing as taught in the Eickemeyer Abstract.

In regard to claim 11, incorporating the rejection of claim 8 (a storage device), it is rejected for the same corresponding reasons put forth in the rejection of claim 4 (a corresponding method).

In regard to claim 12, incorporating the rejection of claim 8 (a storage device), it is rejected for the same corresponding reasons put forth in the rejection of claim 5 (a corresponding method).

In regard to claim 16, incorporating the rejection of claim 15:

> *"...wherein said queue manager is further configured to manage a load queue, a prefetch queue, and a store queue, and wherein said code scheduling module further configured to determine and correlate a maximum desirable number of elements identifier for each of said load queue, said prefetch queue, and said store queue, and further configured to determine a precedence ordering between said load queue, said prefetch queue, and said store queue, and further configured to choose one of said nodes in said ready set that will change the number of elements in one of said load queue, said prefetch queue, or said store queue in accordance with said precedence order and one of said correlated identifiers."*

Claim 16 (an apparatus) is rejected for the same corresponding reasons put forth in the rejection of claim 4 (a corresponding method).

In regard to claim 17, incorporating the rejection of claim 15:

> *"...wherein said queue manager is further configured to determine what number of queues to use in accordance with a target processor and managing said determined number of queues, and wherein said code scheduling module is further configured to, determine a precedence ordering between each of said determined number of queues and is further configured to choose one of said nodes in said ready set that will affect the elements in one of said determined number of queues in accordance with said precedence order."*

Claim 17 (an apparatus) is rejected for the same corresponding reasons put forth in the rejection of claim 5 (a corresponding method).

11. Claims 6, 13, and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rasbold et al., U.S. Patent 5,202,975 as applied to claim 1 above, and further in view of Gutpa et al., U.S. Patent 5,941,983 (hereinafter referred to as Gupta).

In regard to claim 6, incorporating the rejection of claim 1:

"...where said ordering of said instructions in said code file uses a hardware scheduler in accordance with said chosen node and its associated instruction."

Rasbold does not disclose a hardware scheduler, but Gupta discloses a hardware scheduler (column 2, lines 10 – 12). Therefore, it would have been obvious to one skilled in the art at the time the invention was made to modify the Rasbold scheduler with invention with the hardware scheduler of Gupta because the modification provides a means to prevent instructions that are not ready from stalling the machine as taught by Gupta at column 2, lines 55 – 58.

In regard to claim 13, incorporating the rejection of claim 8 (a storage device), it is rejected for the same corresponding reasons put forth in the rejection of claim 6 (a corresponding method).

In regard to claim 18, incorporating the rejection of claim 15:

"...wherein said code scheduler module further comprises a hardware scheduler."

Claim 18 (an apparatus) is rejected for the same corresponding reasons put forth in the rejection of claim 6 (a corresponding method).

Conclusion

12. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure:

U.S. Patent 5,768,594 to Blelloch et al., regarding the scheduling of parallel processors.

13. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Lawrence Shrader whose telephone number is (703) 305-8046. The examiner can normally be reached on M-F 08:00-16:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (703) 305-9662. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Lawrence Shrader
Examiner
Art Unit 2124

16 March 2004

TODD INGBERG
PRIMARY EXAMINER